LECTURE 24

Today

Coding For Interaction

- Setup: Interaction + Coding Architecture
- History
- Tree Codes
- Braverman-Rao Protocol.

Interaction + Error

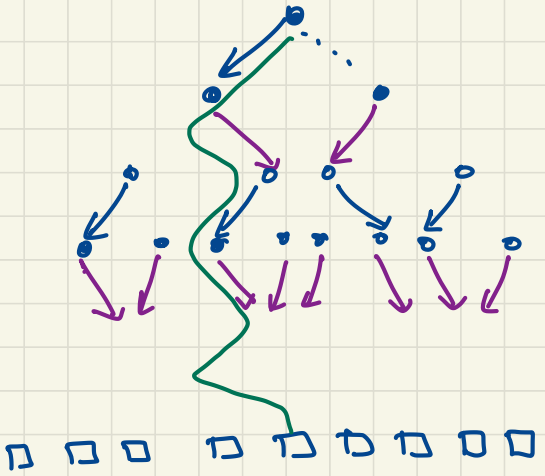
A → B: Please explain Proof of Fermat's Theorem

B → A: Group theory, Galois Groups, Modular Forms Semi-stable Representations (4 lectures)...

A → B: Er.. I meant Fermat's little theorem...

Moral: Few undetected errors easily completely derail interaction.

Formally an interaction = Subgraph of rooted binary tree



Alice's Input = Blue Edges

One child for every node at even depth

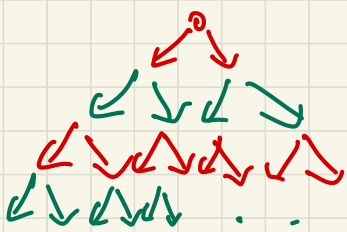
Bob's Input = Purple Edges

one child for every node at odd depth

Goal of interaction: Alice, Bob output leaf obtained by following unique path from root to leaf.

Formally: (Binary) Protocol of length k

- Stage: $V = \{0,1\}^{\leq k} = V_A \cup V_B$
  - V_A: Alice speaks
  - V_B: Bob speaks
- $E = \{(x, x0), (x, x1) \mid x \in \{0,1\}^{<k}\}$
  $= E_A \cup E_B$
  - E_A: source in V_A
  - E_B: source in V_B



- Input (Messages): $\Pi_A \subseteq E_A$: One edge out of every vertex in V_A
  $\Pi_B \subseteq E_B$: "
- Goal: Output valid transcript i.e. path from root (ϕ) to leaf $m \in \{0,1\}^k$ s.t.
  ① m is A-valid (path $\subseteq \Pi_A \cup E_B$)
  ② m is B-valid (path $\subseteq \Pi_B \cup E_A$).
- Obvious: $\exists$ unique valid transcript. Can be determined (by Alice & Bob) with k bits of comm.

Interactive Coding

- $E_A : \Pi_A \mapsto \Gamma_A$ $\quad$ $E_B : \Pi_B \mapsto \Gamma_B$

  (Γ_A, Γ_B) : Edges on $\{0,1\}^{\leq n} \leftarrow F_A \cup F_B$

- functions $f_A, f_B : \{0,1\}^n \to \{0,1\}^k$ $\quad \Gamma_A \subseteq F_A$

- Correct e errors if $\forall\, a, b \in \{0,1\}^n$ s.t. $\Gamma_B \subseteq F_B$

  (1) a is A-valid
  (2) b is B-valid
  } for (Γ_A, Γ_B) $\quad a \in$ path in $\Gamma_A \cup F_B$

  (3) $\Delta(a,b) \leq e$

  we have $f_A(a) = f_B(b) = m$ which is valid for (Γ_A, Γ_B).

History

[Schulman '90]: Introduced problem, "Tree codes",
$\Omega(1)$-Rate protocol correcting $\Omega(1)$-fraction errors.

[Braverman-Rao '11]: $\Omega(1)$-rate protocol correcting $\frac{1}{8}$-fraction errors. "Optimal".

... (many works since).

Tree Codes:

- $T: \{0,1\}^* \to \{0,1\}^*$ s.t.
  - $\forall x, y \quad |x| = |y| \Rightarrow |T(x)| = |T(y)|$
  - $\forall x \quad T(x)$ prefix of $T(x0)$ & $T(x1)$

("online code")

- **Rate**: $\forall k$, $x \in \{0,1\}^k$

$$|T(x)| \leq \frac{k}{R}$$

[today $R = \frac{1}{s}$ for integer n]

$\Rightarrow$ each bit maps to s bits.

$|T(x)| = c \cdot |x|$

x	0	1	0
$T(x)$	000	110	011

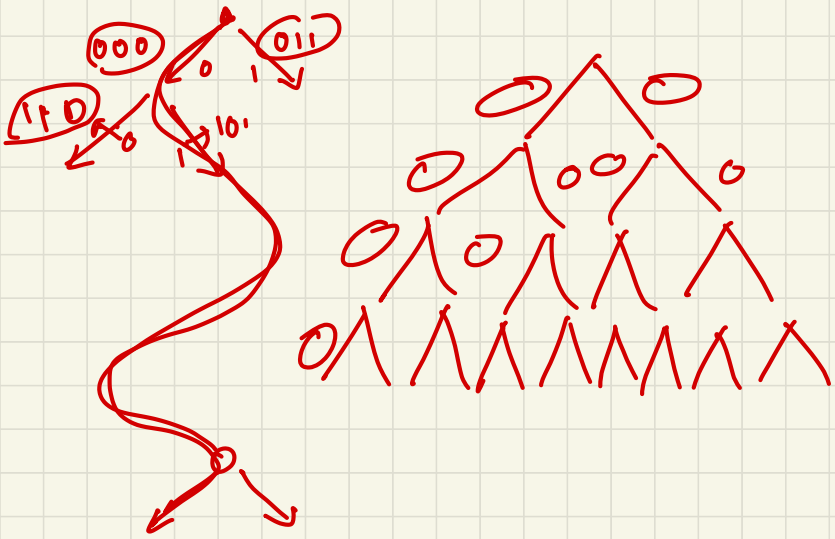
- **distance** δ if $\forall x, y \in \{0,1\}^k$, $i \in [k]$

if $x_i \neq y_i$ then

$$\Delta(T(x), T(y)) \geq \delta \cdot \frac{(k-i+1)}{R}.$$

- **Question**: is $\delta, R = \Omega(1)$ possible?
- **Question**: Is this useful?

Schulman answers Yes & Yes!


000
011
110
10
0
1
0
1

Proof of Existence:

Take T to be a linear code with "upper triangular" generator:

$$\begin{bmatrix} * & * & * & * & * & * & * & * & * & \cdots \\ 0 & 0 & 0 & * & * & * & * & * & * & \\ 0 & 0 & 0 & 0 & 0 & 0 & * & * & * & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\ \vdots & & & & & & & & & \end{bmatrix}$$

Pick $***$'s at random $\Rightarrow$ Toeplitz.

Rate $= \frac{1}{c}$

$\delta = \Omega(1)$?

[with prob > 0]

Exercise

- build G_{2^k} inductively

$$G_{2k} = \begin{bmatrix} G_k & H_k \\ & G_k \end{bmatrix}$$

(H_k: random.)

- Prove $\Pr[G_{2k} \text{ does not work} \mid G_k \text{ works}] = \exp(-k)$
- Conclude G exist.

Exercise: over large alphabets, codes with $\delta \to 1$ exist with $R > 0$

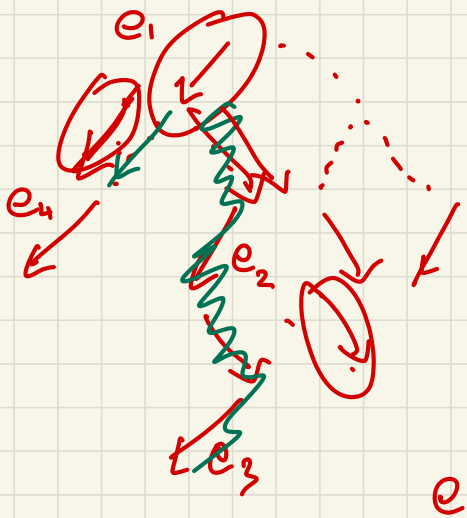
Using Tree Codes

- Schulman: local moves on Π_A, Π_B

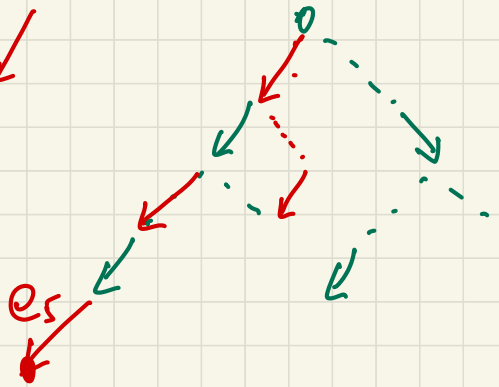
  (roughly ... start proof of FLT ....
  realize mistake, erase board & restart)

- Braverman-Rao. Global moves

  (roughly ... start proof of FLT ....
  realize mistake, don't apologize,
  continue as if everything we discussed
  was still useful.)



$e_1\ e_2\ e_3\ e_4\ e_5$



Braverman Rao: Fix tree codes T^A, T^B (don't have to be same)

Alice's Protocol:

- State = sequence $S_A = (e_1, \dots e_i) \subseteq \Pi_A$
- j^{th} round of communication
  - Recieves message m_j^B; Combines with past to get $m_{\le j}^B$
  - Decodes to sequence $R_{B,j}$ (one that minimizes $\delta(m_{\le j}^B, T^B(R_{B,j}))$
- if successor e to path in $R_{B,j} \cup S_A$ not in S_A then add $e = e_{i+1}$ to S_A.
  [if $R_{B,j}$ looks invalid or successor belongs to E_B, do nothing.]
- Send $(i+1)^{th}$ bit of $T^A(S_A)$ to Bob.
- Repeat for n steps.
- if Path in (Π_A, Π_B) determined output it.

$R_{B,n} \cup S_A$ if it has a unique path to leaf. output.

Analysis (Highlights)

Crux: Measure progress by S_A & S_B = path length $(S_A \cup S_B)$

3 Defns

① $N(i,j)$ = # errors during transmissions i to j in Γ

② $m(i)$ = length of prefix, agreed upon after i transmissions in Γ

③ $t(k)$ = first time that j^{th} edge enters S^A (and is announced)

Key Lemmas

① $m(n) \geq t(k) \Rightarrow$ Done!

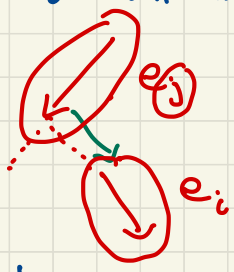
② $m(t(k)-2) < t(k-1)$.

③ $N(m(i)+1, i) \geq \delta \cdot (i - m(i))/2$.

① + ② + ③ $\Rightarrow$ Protocol Corrects $\delta/2$ fraction errors.

Compression

- Can't afford to send $S_A = \{e_1 \dots e_i\}$ at all times.
- Must compress.
- Idea ①:



e_i = e_j followed by two bits.

= express as $(j, b_1 b_2)$.

- Compresses i down to $\log n$ bits.
- Idea ②:

Usually $i - j$ should be small

$\Rightarrow$ express e_i as $(i-j, b_1 b_2)$!

Tree codes : Encoding + Decoding Const. ?

Braverman Rao Protocol · Constructive.

Constructive?

Ideas using randomness?

- No del. efficient protocols?
- Explicit tree-code? Decoding?